

2802Classtype casting

May 12, 2025

0.1 Type Casting

0.1.1 Cast different datatypes to int

```
[ ]: Conclusion- Cast from all types to int is possible but complex and text string  
    ↳ is not possible
```

```
[ ]: we can only pass one argument/parameter for int and float but complex can have  
    ↳ 2 arguments
```

```
[ ]: argument=parameter
```

```
[1]: int(10.0) # Cast to int from float - one argument (10.0)
```

```
[1]: 10
```

```
[2]: int(2.3)
```

```
[2]: 2
```

```
[3]: int(2.3,3.4) # two arguments (2.3 and 3.4)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[3], line 1  
----> 1 int(2.3,3.4)  
  
TypeError: 'float' object cannot be interpreted as an integer
```

```
[4]: int(10+20j) # cannot cast complex to int
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[4], line 1  
----> 1 int(10+20j)
```

```
TypeError: int() argument must be a string, a bytes-like object or a real
↳number, not 'complex'
```

```
[5]: int("10") # cast from number string to int is possible but text string is not
```

```
[5]: 10
```

```
[6]: int("ten") # cast from number string to int is possible but text string is not
```

```
-----
ValueError                                Traceback (most recent call last)
Cell In[6], line 1
----> 1 int("ten")

ValueError: invalid literal for int() with base 10: 'ten'
```

```
[7]: int(True) # Boolean to int is 1
```

```
[7]: 1
```

```
[8]: int(False)
```

```
[8]: 0
```

```
[9]: Int(True+True) ## cannot pass two arguments
```

```
-----
NameError                                Traceback (most recent call last)
Cell In[9], line 1
----> 1 Int(True+True)

NameError: name 'Int' is not defined
```

```
[11]: True # not casting its boolean
```

```
[11]: True
```

```
[12]: True+True
```

```
[12]: 2
```

```
[13]: False+False
```

```
[13]: 0
```

0.2 cast to float from other data forms

```
[15]: float(10)
```

```
[15]: 10.0
```

```
[16]: float(True)
```

```
[16]: 1.0
```

```
[18]: float(False)
```

```
[18]: 0.0
```

```
[19]: float(10,20) ## cannot apss 2 arguments
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[19], line 1  
----> 1 float(10,20)  
  
TypeError: float expected at most 1 argument, got 2
```

```
[20]: float(10+20j)
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[20], line 1  
----> 1 float(10+20j)  
  
TypeError: float() argument must be a string or a real number, not 'complex'
```

```
[21]: float("10")
```

```
[21]: 10.0
```

```
[22]: float("Ten")
```

```
-----  
ValueError                                Traceback (most recent call last)  
Cell In[22], line 1  
----> 1 float("Ten")  
  
ValueError: could not convert string to float: 'Ten'
```

0.3 cast to Complex from other data forms

```
[57]: z=real+imaginary*1j  
      z=3+2j
```

```
-----  
NameError                                Traceback (most recent call last)  
Cell In[57], line 1  
----> 1 z=real+imaginary*1j  
      2 z=3+2j  
  
NameError: name 'real' is not defined
```

```
[60]: z=3+4j  
      print(z.real)  
      print(z.imag)
```

```
3.0  
4.0
```

```
[67]: a=3+4j  
      b=5+6j  
      print(a+b)  
      print(a-b)  
      print(a*b)  
      print(a/b)
```

```
(8+10j)  
(-2-2j)  
(-9+38j)  
(0.6393442622950819+0.03278688524590165j)
```

```
[78]: z=3+4j  
      print(abs(z))  
      print(z.conj())
```

```
5.0
```

```
-----  
AttributeError                            Traceback (most recent call last)  
Cell In[78], line 3  
      1 z=3+4j  
      2 print(abs(z))  
----> 3 print(z.conj())  
  
AttributeError: 'complex' object has no attribute 'conj'
```

[]:

[61]: `complex(10)`

[61]: (10+0j)

[62]: `complex(10.5)`

[62]: (10.5+0j)

[63]: `complex(10,20)`

[63]: (10+20j)

[29]: `complex(True)`

[29]: (1+0j)

[30]: `complex(False)` *##0+0j is 0j*

[30]: 0j

[31]: `complex(True+True)`

[31]: (2+0j)

[32]: `complex(True,True)`

[32]: (1+1j)

[33]: `complex(10,20,30)` *## can pass only 2 arguments*

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[33], line 1  
----> 1 complex(10,20,30)  
  
TypeError: complex() takes at most 2 arguments (3 given)
```

[34]: `complex("10")`

[34]: (10+0j)

[36]: `complex("10","20")`

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[36], line 1  
----> 1 complex("10","20")  
  
TypeError: complex() can't take second arg if first is a string
```

```
[38]: complex("Ten")
```

```
-----  
ValueError                                Traceback (most recent call last)  
Cell In[38], line 1  
----> 1 complex("Ten")  
  
ValueError: complex() arg is a malformed string
```

cast to String from other datatypes

```
[39]: str(10)
```

```
[39]: '10'
```

```
[40]: str(10,20) ## int can only have one argument
```

```
-----  
TypeError                                Traceback (most recent call last)  
Cell In[40], line 1  
----> 1 str(10,20)  
  
TypeError: str() argument 'encoding' must be str, not int
```

```
[41]: str(10.0)
```

```
[41]: '10.0'
```

```
[42]: str(10+20j)
```

```
[42]: '(10+20j)'
```

```
[43]: str(True)
```

```
[43]: 'True'
```

```
[44]: str(False)
```

```
[44]: 'False'
```

```
[45]: str(0)
```

```
[45]: '0'
```

```
[46]: str('ten')
```

```
[46]: 'ten'
```

0.4 cast to boolean from other datatypes

```
[47]: bool(10)
```

```
[47]: True
```

```
[48]: bool(10.5)
```

```
[48]: True
```

```
[49]: bool(0)
```

```
[49]: False
```

```
[50]: bool(10+20j)
```

```
[50]: True
```

```
[51]: bool('Ten')
```

```
[51]: True
```

```
[ ]: bool("T
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```

```
[ ]:
```